

MCMC Methods for Integer Least-Squares Problems

Babak Hassibi*, Alexandros G. Dimakis†, and Dimitris Papailiopoulos†

*California Institute of Technology, Department of Electrical Engineering,
Pasadena, CA 91125, USA
Email: hassibi@caltech.edu

†University of Southern California, Department of Electrical Engineering - Systems,
Los Angeles, CA 90089-2560
E-mail: adim,papailio@usc.edu

Abstract—We consider the problem of finding the least-squares solution to a system of linear equations where the unknown vector has integer entries (or, more precisely, has entries belonging to a subset of the integers), yet where the coefficient matrix and given vector are comprised of real numbers. Geometrically, this problem is equivalent to finding the closest lattice point to a given point and is known to be NP hard. In communication applications, however, the given vector is not arbitrary, but is a lattice point perturbed by some noise vector. Therefore it is of interest to study the computational complexity of various algorithms as a function of the noise variance or, often more appropriately, the SNR.

In this paper, we apply a particular version of the Monte Carlo Markov chain (MCMC) approach to solving this problem, which is called a “heat bath”. We show that there is a trade-off between the mixing time of the Markov chain (how long it takes until the chain reaches its stationary distribution) and how long it takes for the algorithm to find the optimal solution once the chain has mixed. The complexity of the algorithm is essentially the sum of these two times. More specifically, the higher the temperature, the faster the mixing, yet the slower the discovery of the optimal solution in steady state. Conversely, the lower the temperature, the slower the mixing, yet the faster the discovery of the optimal solution once the chain is mixed.

We first show that for the probability of error of the maximum-likelihood (ML) solution to go to zero the SNR must scale at least as $2 \ln N + \alpha(N)$, where N is the ambient problem dimension and $\alpha(N)$ is any sequence that tends to positive infinity. We further obtain the optimal value of the temperature such that the average time required to encounter the optimal solution in steady state is polynomial. Simulations show that, with this choice of the temperature parameter, the optimal solution can be found in reasonable time. This suggests that the Markov chain mixes in polynomial-time, though we have not been able to prove this. It seems reasonable to conjecture that for SNR scaling as $O((\ln(N))^{1+\epsilon})$, and for appropriate choice of the temperature parameter, the heat bath algorithm finds the optimal solution in polynomial-time.¹

I. INTRODUCTION

The problem of performing Maximum Likelihood (ML) decoding in digital communication has gained much attention

¹This work was supported in part by the National Science Foundation under grants CCF-0729203, CNS-0932428 and CCF-1018927, by the Office of Naval Research under the MURI grant N00014-08-1-0747, and by Caltech’s Lee Center for Advanced Networking. Many of the results of this paper were first reported in [1].

over the years. One method to obtain the ML solution is Sphere Decoding (SD) [2]–[6]. Over a wide range of Signal-to-Noise Ratios (SNR)s the average complexity of SD is significantly smaller than exhaustive search detectors, but in worst case the complexity is still exponential [7]. Thus, in scenarios with poor SNR or in Multiple-Input Multiple-Output (MIMO) systems with huge transmit and receive dimensions, even SD can be infeasible. A way to overcome this problem is to use approximate Markov Chain Monte Carlo (MCMC) detectors instead, which asymptotically can provide the optimal solution, [8], [9]. The “heat bath” (also known as Glauber dynamics) is one MCMC method, which is used for sampling from distributions of multiple dimensions. This method has, among others, been proposed for detection purposes in wireless communication in [10]–[13] (see also the references therein). In our earlier work, [1], we described and analyzed a new way of solving the integer least-squares problem using MCMC. It was shown that the method can be used for achieving a near-optimal and computationally efficient solution of the problem, even for systems having a huge dimension.

In this paper we review the main results of [1] and discuss some of the complexity issues. Simulation results suggest that in certain regimes of the SNR the optimal solution can be found quite efficiently. However, we have been unable to theoretically pinpoint this regime and the main problem of determining the expected complexity of the algorithm remains unresolved.

II. SYSTEM MODEL

We consider a real-valued block-fading MIMO antenna system, with N transmit and N receive dimensions, with known channel coefficients.² The received signal $\mathbf{y} \in \mathbb{R}^N$ can be expressed as

$$\mathbf{y} = \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \mathbf{s} + \mathbf{v}, \quad (1)$$

²For simplicity we have assumed that the receive and the transmit dimensions are the same, but the results presented in the paper can be generalized to cover different dimensions.

where $\mathbf{s} \in \Omega^N$ is the transmitted signal, and Ω denotes the constellation set. To simplify the derivations in the paper we will assume that $\Omega = \{\pm 1\}$, although more complicated constellations can be readily analyzed in the same fashion. $\mathbf{v} \in \mathbb{R}^N$ is the noise vector where each entry is Gaussian $\mathcal{N}(0, 1)$ and independent identically distributed (i.i.d.), and $\mathbf{H} \in \mathbb{R}^{N \times N}$ denotes the channel matrix with i.i.d. $\mathcal{N}(0, 1)$ entries.³ The normalization in (1) guarantees that SNR represents the signal-to-noise ratio per receive dimension (which we define as the ratio of the total transmit energy per channel use divided by the per-component noise variance as described in among others [6]). As explained further below, for analysis purposes we will focus on the regime where $(\text{SNR} - 2 \ln(N)) \rightarrow \infty$, in order to get the probability of error of the ML detector to go to zero. Further, in our analysis, without loss of generality, we will assume that the all minus one vector was transmitted, $\mathbf{s} = -\mathbf{1}$. Therefore

$$\mathbf{y} = \mathbf{v} - \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \mathbf{1}. \quad (2)$$

We are considering a minimization of the average error probability $P(\mathbf{e}) \triangleq P(\hat{\mathbf{s}} \neq \mathbf{s})$, which is obtained by performing Maximum Likelihood Sequence Detection (here simply referred to as ML detection) given by

$$\hat{\mathbf{s}} = \arg \min_{\mathbf{s} \in \Omega^N} \left\| \mathbf{y} - \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \mathbf{s} \right\|^2. \quad (3)$$

III. THE HEAT BATH METHOD

One way of solving the optimization problem given in (3) is by using Markov Chain Monte Carlo (MCMC) simulations. However, rather than insist that the Markov chain asymptotically converge to the optimal solution [14], we will insist that its steady-state distribution be one that will encounter the optimal solution with non-negligible frequency (one over a polynomial in N). More specifically, the MCMC detector we investigate here is called a heat bath. At any time instant k , say, the algorithm randomly chooses an entry of the vector \mathbf{s} , the j -th entry, say, and adopts the value $\omega \in \Omega$ with probability

$$p(\hat{\mathbf{s}}_j^{(k)} = \omega | \theta) = \frac{e^{-\frac{1}{2\alpha^2} \left\| \mathbf{y} - \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \tilde{\mathbf{s}}_{j|\omega} \right\|^2}}{\sum_{\tilde{\mathbf{s}}_{j|\omega} \in \Omega} e^{-\frac{1}{2\alpha^2} \left\| \mathbf{y} - \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \tilde{\mathbf{s}}_{j|\omega} \right\|^2}}, \quad (4)$$

where $\tilde{\mathbf{s}}_{j|\omega}^T \triangleq [\hat{\mathbf{s}}_{1:j-1}^{(k)}, \omega, \hat{\mathbf{s}}_{j+1:N_T}^{(k-1)}]^T$ and where we for simplicity have introduced $\theta = \{\hat{\mathbf{s}}^{(k-1)}, \mathbf{y}, \mathbf{H}\}$.⁴ α represents a tunable positive parameter which controls the mixing time of the Markov chain, this parameter is also sometimes called

the "temperature". The larger α is the faster the mixing time of the Markov chain will be, but as we will show in the paper, there is an upper limit on α , in order to ensure that the probability of finding the optimal solution in steady state is not exponentially small. The MCMC method will with probability $p(\hat{\mathbf{s}}_j^{(k)} = \omega | \theta)$ keep ω at the j 'th index in estimated symbol vector, and compute conditional probability the $(j+1)$ th index in a similar fashion. We define one iteration of the Gibbs sampler as a randomly-ordered update of all the $j = \{1, \dots, N_T\}$ indices in the estimated symbol vector $\hat{\mathbf{s}}$.⁵ The initialization of the symbol vector $\hat{\mathbf{s}}^{(0)}$ can either be chosen randomly or, alternatively, e.g. the zero-forcing solution can be used.

A. Complexity of the heat bath

The conditional probability for the j 'th symbol in (4) can be computed efficiently by reusing the result obtained for the $j-1$ 'th symbol, when we evaluate $\left\| \mathbf{y} - \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \tilde{\mathbf{s}}_{j|\omega} \right\|^2$. Since we are only changing the j 'th symbol in the symbol vector, the difference $\mathbf{d}_j \triangleq \mathbf{y} - \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \tilde{\mathbf{s}}_{j|\omega}$ can be expressed as

$$\mathbf{d}_j = \mathbf{d}_{j-1} - \sqrt{\frac{\text{SNR}}{N}} \mathbf{H}_{1:N,j} \Delta s_{j|\omega}, \quad (5)$$

where $\Delta s_{j|\omega} \triangleq s_{j|\omega}^{(k)} - s_{j|\omega}^{(k-1)}$. Thus, the computation of conditional probability of certain symbol in the j 'th position costs $2N$ operations, where we define an operation as a Multiply and Accumulate (MAC) instruction.⁶ This leads to a complexity of $\mathcal{O}(2N^2[|\Omega| - 1])$ operations per iteration. For further details on the implementation see [15].

IV. PROBABILITY OF ERROR

In this paper, we are interested in evaluating the performance of the aforementioned heat bath, compared to the ML solution. To ease our analysis, we will assume that the ML detector finds the correct transmitted vector. Before we derive the probability of error for the ML detector, we will state a lemma which we will make repeated use of.

Lemma IV.1 (Gaussian Integral). *Let \mathbf{v} and \mathbf{x} be independent Gaussian random vectors with distribution $\mathcal{N}(\mathbf{0}, \mathbf{I}_N)$ each. Then, if $1 - 2a^2\eta(1 + 2\eta) > 0$,*

$$E \left\{ e^{\eta(\|\mathbf{v} + a\mathbf{x}\|^2 - \|\mathbf{v}\|^2)} \right\} = \left(\frac{1}{1 - 2a^2\eta(1 + 2\eta)} \right)^{N/2}. \quad (6)$$

³While this last assumption may not hold for many channel matrices \mathbf{H} it allows us to push through a closed-form analysis of the problem and algorithm quite a way.

⁴When we compute the probability of symbol ω at the j 'th position, we more precisely condition on the symbols $\hat{\mathbf{s}}_{1:j-1}^{(k)}$ and $\hat{\mathbf{s}}_{j+1:N_T}^{(k-1)}$, but to keep the notation simple, we do not explicitly state that in the equations above.

⁵We need a randomly-ordered update for the Markov chain to be reversible and for our subsequent analysis to go through. It is also possible to just randomly select a symbol j to update, without insisting that a full sequence be done. This also makes the Markov chain reversible and has the same steady state distribution. In practice a fixed, say sequential, order can be employed, although the Markov chain is no longer reversible. Note that our theoretical analysis is assuming randomly selected symbol updates for analytical convenience. In our experimental section we used a sequential updating order which empirically yields a slight convergence acceleration.

⁶We need to compute both the inner product $\mathbf{d}_j^T \mathbf{d}_j$ and the product $\mathbf{H}_{1:N,j} \Delta s_{j|\omega}$.

Proof: See Appendix IX-A for a detailed proof. ■

Assuming that the vector $\mathbf{s} = -\mathbf{1}$ was transmitted, the ML detector will make an error if there exists a vector $\mathbf{s} \neq -\mathbf{1}$ such that

$$\left\| \mathbf{y} - \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \mathbf{s} \right\|^2 \leq \left\| \mathbf{y} + \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \mathbf{1} \right\|^2 = \|\mathbf{v}\|^2.$$

In other words,

$$\begin{aligned} P_e &= \text{Prob} \left(\left\| \mathbf{y} - \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \mathbf{s} \right\|^2 \leq \|\mathbf{v}\|^2 \right) \\ &= \text{Prob} \left(\left\| \mathbf{v} + \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} (-\mathbf{1} - \mathbf{s}) \right\|^2 \leq \|\mathbf{v}\|^2 \right), \end{aligned}$$

for some $\mathbf{s} \neq -\mathbf{1}$, which can be formulated as

$$P_e = \text{Prob} \left(\left\| \mathbf{v} + 2\sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \boldsymbol{\delta} \right\|^2 \leq \|\mathbf{v}\|^2 \right),$$

for some $\boldsymbol{\delta} \neq \mathbf{0}$. Note that in the above equation $\boldsymbol{\delta}$ is a vector of zeros and -1 's. Now using the union bound

$$P_e \leq \sum_{\boldsymbol{\delta} \neq \mathbf{0}} \text{Prob} \left(\left\| \mathbf{v} + 2\sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \boldsymbol{\delta} \right\|^2 \leq \|\mathbf{v}\|^2 \right). \quad (7)$$

We will use the Chernoff bound to bound the quantity inside the summation. Thus,

$$\text{Prob} \left(\left\| \mathbf{v} + 2\sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \boldsymbol{\delta} \right\|^2 \leq \|\mathbf{v}\|^2 \right) \quad (8a)$$

$$\leq E \left\{ e^{-\beta \left(\left\| \mathbf{v} + 2\sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \boldsymbol{\delta} \right\|^2 - \|\mathbf{v}\|^2 \right)} \right\} \quad (8b)$$

$$= \left(\frac{1}{1 + 8 \frac{\text{SNR} \|\boldsymbol{\delta}\|^2}{N} \beta (1 - 2\beta)} \right)^{N/2}, \quad (8c)$$

where $\beta \geq 0$ is the Chernoff parameter, and where we have used Lemma IV.1 with $\eta = -\beta$ and $a = 2\sqrt{\frac{\text{SNR} \|\boldsymbol{\delta}\|^2}{N}}$, since

$$E \left\{ \left(2\sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \boldsymbol{\delta} \right) \left(2\sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \boldsymbol{\delta} \right)^* \right\} = 4 \frac{\text{SNR} \|\boldsymbol{\delta}\|^2}{N} \mathbf{I}_N.$$

The optimal value for β is $\frac{1}{4}$, which yields the tightest bound

$$\text{Prob} \left(\left\| \mathbf{v} + 2\sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \boldsymbol{\delta} \right\|^2 \leq \|\mathbf{v}\|^2 \right) \leq \left(\frac{1}{1 + \frac{\text{SNR} \|\boldsymbol{\delta}\|^2}{N}} \right)^{N/2}. \quad (9)$$

Note that this depends only on $\|\boldsymbol{\delta}\|^2$, the number of nonzero entries in $\boldsymbol{\delta}$. Plugging this into the union bound yields

$$P_e \leq \sum_{i=1}^N \binom{N}{i} \left(\frac{1}{1 + \frac{\text{SNR}_i}{N}} \right)^{N/2}. \quad (10)$$

Let us first look at the linear (i.e., i proportional to N) terms in the above sum. Thus,

$$\binom{N}{i} \left(\frac{1}{1 + \frac{\text{SNR}_i}{N}} \right)^{N/2} \approx e^{NH(\frac{i}{N}) - \frac{N}{2} \ln \left(1 + \frac{\text{SNR}_i}{N} \right)},$$

where $H(\cdot)$ is entropy in “nats”. Clearly, if $\lim_{N \rightarrow \infty} \text{SNR} = \infty$, then the linear terms go to zero (superexponentially fast).

Let us now look at the sublinear terms. In particular, let us look at $i = 1$:

$$N \left(\frac{1}{1 + \frac{\text{SNR}}{N}} \right)^{N/2} \approx N e^{-\text{SNR}/2}.$$

Clearly, to have this term go to zero, we require that $(\text{SNR} - 2 \ln N) \rightarrow \infty$. A similar argument shows that all other sublinear terms also go to zero, and so.⁷

Lemma IV.2 (SNR scaling). *If $(\text{SNR} - 2 \ln N) \rightarrow \infty$, then $P_e \rightarrow 0$ as $N \rightarrow \infty$.*

V. COMPUTING THE OPTIMAL α

Assuming that the vector $\mathbf{s} = -\mathbf{1}$ has been transmitted, the probability of finding this solution *after the Markov chain has mixed* is simply π_{-1} , the steady-state probability of being in the all -1 state. Clearly, if this probability is exponentially small, it will take exponentially long for the heat bath to find it. We will therefore insist that the mean of π_{-1} be only polynomially small.⁸

A. Mean of π_{-1}

This calculation has a lot in common with the one given in Section IV. Note that the steady state value of π_{-1} is simply

$$\pi_{-1} = \frac{e^{-\frac{1}{2\alpha^2} \left\| \mathbf{y} + \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \mathbf{1} \right\|^2}}{\sum_{\mathbf{s}} e^{-\frac{1}{2\alpha^2} \left\| \mathbf{y} + \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \mathbf{s} \right\|^2}} \quad (11a)$$

$$= \frac{e^{-\frac{1}{2\alpha^2} \|\mathbf{v}\|^2}}{\sum_{\mathbf{s}} e^{-\frac{1}{2\alpha^2} \left\| \mathbf{v} + \sqrt{\frac{\text{SNR}}{N}} \mathbf{H} (\mathbf{s} - \mathbf{1}) \right\|^2}} \quad (11b)$$

$$= \frac{e^{-\frac{1}{2\alpha^2} \|\mathbf{v}\|^2}}{\sum_{\boldsymbol{\delta}} e^{-\frac{1}{2\alpha^2} \left\| \mathbf{v} + 2\sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \boldsymbol{\delta} \right\|^2}} \quad (11c)$$

$$= \frac{1}{\sum_{\boldsymbol{\delta}} e^{-\frac{1}{2\alpha^2} \left(\left\| \mathbf{v} + 2\sqrt{\frac{\text{SNR}}{N}} \mathbf{H} \boldsymbol{\delta} \right\|^2 - \|\mathbf{v}\|^2 \right)}}, \quad (11d)$$

⁷Due to space constraints we only present a sketch of this bound. A rigorous proof can be given using the saddle point method, similarly to the proof in the next section.

⁸We remark that we will take this approach, rather than insist that the Markov chain converge to the optimal solution simply because the latter would require a very small value of α and hence slow down the mixing time considerably. In effect, we are looking for the largest value of α (so that the mixing time is as small as possible) so that the optimal solution can be found in polynomial time in steady-state.

where δ is a vector of zeros and ones and the summations (over s and δ) are over 2^n terms.

Now, by Jensen's inequality

$$E\{\pi_{-1}\} \geq \frac{1}{E\left\{\frac{1}{\pi_{-1}}\right\}} \quad (12a)$$

$$= \frac{1}{E\left\{\sum_{\delta} e^{-\frac{1}{2\alpha^2}\left(\left\|\mathbf{v} + 2\sqrt{\frac{\text{SNR}}{N}}\mathbf{H}\delta\right\|^2 - \|\mathbf{v}\|^2\right)}\right\}} \quad (12b)$$

$$= \frac{1}{\sum_{\delta} E\left\{e^{-\frac{1}{2\alpha^2}\left(\left\|\mathbf{v} + 2\sqrt{\frac{\text{SNR}}{N}}\mathbf{H}\delta\right\|^2 - \|\mathbf{v}\|^2\right)}\right\}} \quad (12c)$$

$$= \frac{1}{1 + \sum_{\delta \neq 0} \left(\frac{1}{1 + 4\frac{\text{SNR}\|\delta\|^2}{N}\frac{1}{\alpha^2}(1 - \frac{1}{\alpha^2})}\right)^{N/2}} \quad (12d)$$

$$= \frac{1}{1 + \sum_{i=1}^N \binom{N}{i} \left(\frac{1}{1 + \frac{\beta i}{N}}\right)^{N/2}}. \quad (12e)$$

In (12d) we have used Lemma IV.1 and in (12e) we have defined $\beta \triangleq 4\text{SNR}\frac{1}{\alpha^2}(1 - \frac{1}{\alpha^2})$. While it is possible to focus on the linear and sublinear terms in the above summation separately, to give conditions for $E\{\pi_{-1}\}$ to have the form of $1/\text{poly}(N)$, we will be interested in the exact exponent and so will need a more accurate estimate. To do this we shall use saddle point integration. Note that

$$\binom{N}{i} \left(\frac{1}{1 + \frac{\beta i}{N}}\right)^{N/2} \approx e^{NH(\frac{i}{N}) - \frac{N}{2} \ln(1 + \frac{\beta i}{N})},$$

where again $H(\cdot)$ represents the entropy in "nats". And so the summation in the denominator of (12e) can be approximated as a Stieltjes integral:

$$\sum_{i=1}^N \binom{N}{i} \left(\frac{1}{1 + \frac{\beta i}{N}}\right)^{N/2} \approx N \sum_{i=1}^N e^{NH(\frac{i}{N}) - \frac{N}{2} \ln(1 + \frac{\beta i}{N})} \frac{1}{N} \quad (13a)$$

$$\approx N \int_0^1 e^{NH(x) - \frac{N}{2} \ln(1 + \beta x)} dx. \quad (13b)$$

For large N , this is a saddle point integral and can be approximated by the formula

$$\int_0^1 e^{Nf(x)} dx \approx \sqrt{\frac{2\pi}{N|f''(x_0)|}} e^{Nf(x_0)}, \quad (14)$$

where x_0 , is the saddle point of $f(\cdot)$, i.e., $f'(x_0) = 0$. In our case,

$$f(x) = -x \ln x - (1-x) \ln(1-x) - \frac{1}{2} \ln(1 + \beta x),$$

and so

$$f'(x) = \ln \frac{1-x}{x} - \frac{1}{2} \frac{\beta}{1 + \beta x}.$$

In general, it is not possible to solve for $f'(x_0) = 0$ in closed form. However, in our case, if we assume that $\beta = 4\text{SNR}\frac{1}{\alpha^2}(1 - \frac{1}{\alpha^2}) \gg 1$ (which is true since the SNR grows at least logarithmically), then it is not too hard to verify that the saddle point is given by

$$x_0 = e^{-\frac{\beta}{2}}. \quad (15)$$

And hence $f(x_0) =$

$$\begin{aligned} & -e^{-\frac{\beta}{2}} \ln e^{-\frac{\beta}{2}} - (1 - e^{-\frac{\beta}{2}}) \ln(1 - e^{-\frac{\beta}{2}}) - \frac{1}{2} \ln(1 + \beta e^{-\frac{\beta}{2}}) \\ & \approx \frac{\beta}{2} e^{-\frac{\beta}{2}} + e^{-\frac{\beta}{2}} - \frac{1}{2} \beta e^{-\frac{\beta}{2}} = e^{-\frac{\beta}{2}}, \end{aligned}$$

and further plugging x_0 into $f''(x) = -\frac{1}{x} - \frac{1}{1-x} - \frac{1}{2} \frac{\beta^2}{(1+\beta x)^2}$, yields

$$f''(x_0) \approx -e^{\frac{\beta}{2}} - 1 + \frac{1}{2} \beta^2 \approx -e^{\frac{\beta}{2}}. \quad (16)$$

Replacing these into the saddle point expression in (14) show that

$$\sum_{i=1}^N \binom{N}{i} \left(\frac{1}{1 + \frac{\beta i}{N}}\right)^{N/2} \approx \sqrt{2\pi/N} \exp\left(Ne^{-\frac{\beta}{2}} - \frac{\beta}{4}\right). \quad (17)$$

We want $E\{\pi_{-1}\}$ to behave as $\frac{1}{N^\zeta}$ and according to (12) this means that we want the expression in (17) to behave as N^ζ . Let us take

$$e^{Ne^{-\frac{\beta}{2}}} = N^\zeta.$$

Solving for β yields

$$\beta = 4\text{SNR} \frac{1}{\alpha^2} \left(1 - \frac{1}{\alpha^2}\right) = 2(\ln N - \ln \ln N - \ln \zeta). \quad (18)$$

Incidentally, this choice of β yields $e^{-\frac{\beta}{4}} \approx \frac{1}{\sqrt{N}}$, and so we have the following result.

Lemma V.1 (Mean of π_{-1}). *If α is chosen such that*

$$\frac{\alpha^2}{1 - \frac{1}{\alpha^2}} = \frac{2\text{SNR}}{\ln N - \ln \ln N - \ln \zeta}, \quad (19)$$

then

$$E\{\pi_{-1}\} \geq N^{-\zeta}. \quad (20)$$

B. Value of α

Note that from (12e) it is clear that the larger β is, the larger π_{-1} is. Therefore, the range of α that gives a polynomially small probability to π_{-1} is

$$\frac{\alpha^2}{1 - \frac{1}{\alpha^2}} \geq \frac{2\text{SNR}}{\ln N - \ln \ln N - \ln \zeta}. \quad (21)$$

It can be shown that in the regime, $\text{SNR} > 2 \ln N$, the above quadratic inequality in α has two positive real solutions, $\alpha_+ \geq \alpha_-$, and that the inequality holds for all $\alpha \in [\alpha_-, \alpha_+]$.

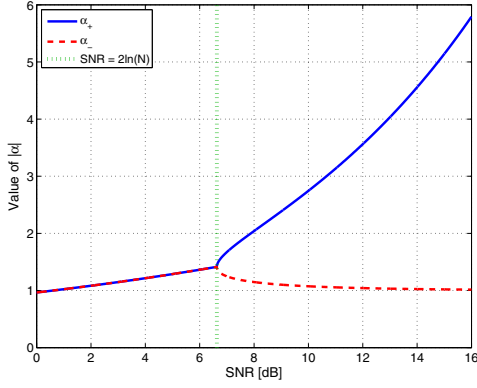


Figure 1: Value of α vs. SNR for system size $N = 10$.

We know that, the larger α is, the faster the Markov chain mixes.⁹ Therefore it is reasonable that we choose the largest permissible value for α , i.e., α_+ .

Figures 1 and 2 show the values of α_+ and α_- as a function of SNR for systems with $N = 10$ and $N = 50$, when we have $\zeta = 1/\ln(N)$.

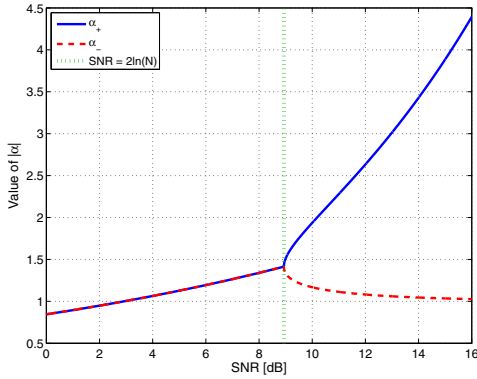


Figure 2: Value of α vs. SNR for system size $N = 50$.

VI. MIXING TIME OF MARKOV CHAIN

Of course, the open question we have so far ignored is whether the Markov chain is rapidly mixing when using the strategy above for choosing α . The simulations, which we shall momentarily present in Section VII seem to indicate that this is the case. In any event, the complexity of the algorithm is roughly given by

$$\text{mixing time} + \text{time to find optimal solution in steady state} \quad (22)$$

As mentioned earlier, increasing α reduces the mixing time, yet increases $\frac{1}{\pi_{-1}}$. Reducing α has the opposite effect.

There exist standard methods for bounding the mixing time of Markov chains. These include:

⁹In general, there is a trade-off between faster mixing time of the Markov chain (due to an increase of α) versus slower encountering the optimal solution in steady-state. In fact, at infinite temperature our algorithm reduces to a random walk in a hypercube which mixes in $O(N \ln N)$ time.

- 1) *The Poincare method*: which is basically a flow argument where one attempts to “guess” the solution to a linear program
- 2) *Cheeger’s inequality*: which is essentially a cut-set bound (involving the probability that the Markov chain transitions from one set of states to its complementary set)
- 3) *Coupling*: based on the analysis of two copies of the Markov chain initialized from “close” states

Our current analysis using the Poincare and Cheeger methods yields polynomial-time mixing only when the SNR scales as $O(\frac{N}{(\ln N)^a})$, which is too high.

In the coupling argument, if two copies of the MC are initialized with states that differ only in one bit and the states of the system remain close (i.e., are coupled for all time), then one can bound the mixing time of the original chain with how close the two states remain over time. Using such a coupling argument, we have been able to show that if the heat bath is initialized with a state vector s that has only $O((\ln N)^a)$ errors, then it converges in polynomial time. However, how to initialize the chain with such an initial s remains unclear. One can do so with straightforward methods if, again, the SNR scales as $O(\frac{N}{(\ln N)^a})$, which as mentioned before, remains too high.

A. Analysis using a slower Markov chain

It seems the aforementioned bounds are too loose because the methods are too general and we have not been able to fully exploit the structure of the Markov chain.

One alternative is to reduce the original MC to a slower MC with N , rather than 2^N , states. To do so, one can lump all state vectors s with the same Hamming weight together into a single state. Of course, there is severe loss of information in this reduction. However, to guarantee that the reduced Markov chain mixes slower than the original one, we can determine its transition probabilities by the minimum transition probabilities (from one set of states with a certain Hamming distance to another such set) in the original MC. The reduced MC has a simple linear topology (since the Hamming weights can differ by at most one in each transition) and if the transition probabilities decay polynomially then it, and the original MC, mix in polynomial time.

We conjecture that if $SNR = O((\ln N)^a)$, for $a > 1$, then the chain mixes in polynomial-time and the MCMC algorithm finds the optimal solution in polynomial-time.

VII. SIMULATION RESULTS

In this section we present simulation results for a MIMO $N \times N$ system with a full square channel matrix containing i.i.d. Gaussian entries. In Fig. 3 and Fig. 4 the Bit Error Rate (BER) of the heat bath, initialized with a random s , has been evaluated as a function of the number of iterations in a 10×10 system using a variety of α values. Thereby, we can inspect how the parameter α affects the convergence rate of the heat bath. The performance of the Maximum Likelihood (ML), the Zero-Forcing (ZF), and the Linear Minimum Mean Square

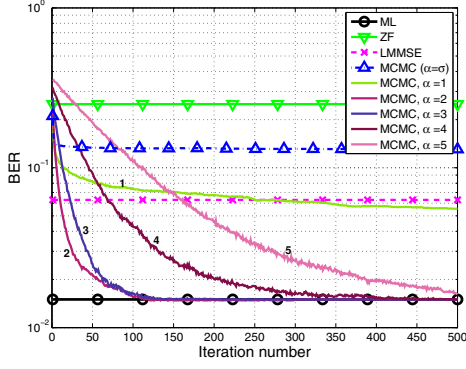


Figure 3: BER vs. iterations, 10×10 . SNR = 10 dB.

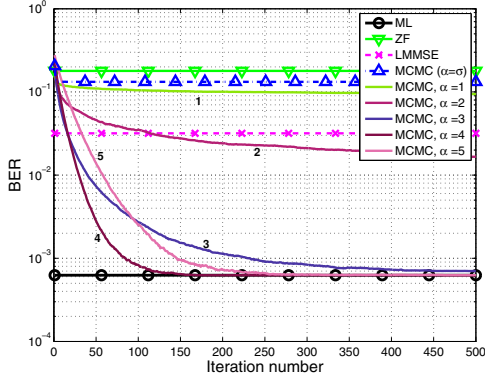


Figure 4: BER vs. iterations, 10×10 system. SNR = 14 dB.

Error (LMMSE) detector has also been plotted, to ease the comparison of the heat bath with these. It is seen that the heat bath outperforms both the ZF and the LMMSE detector after only a few iterations in all the presented simulations, when the tuning parameter α is chosen properly. Furthermore, it is observed that the parameter α has a huge influence on the convergence rate and that the heat bath converges toward the ML solution as a function of the iterations.¹⁰ The optimal value of α (in terms of convergence rate) is quite close to the theoretical values from Fig. 1 of $\alpha_+ = 2.7$ and $\alpha_+ = 4.6$ at SNR's at 10 and 14dB, respectively. It is also observed that the performance of the heat bath is significantly deteriorated if the temperature parameter is chosen based on the SNR (and thereby on the noise variance), such that $\alpha = \sigma \triangleq 1/\text{SNR}$. Thus, the latter strategy is clearly not a wise choice.

Figure 5 shows the BER performance for the MCMC detector for fixed number of iterations, $k = 100$. From the figure we see that the SNR has a significant influence on the optimal choice of α given a fixed number of iterations.

The performance of the heat bath is also shown for a 50×50 system, which represents a ML decoding problem of huge complexity where an exhaustive search would require $2^{50} \approx 10^{15}$ evaluations. For this problem even the sphere

¹⁰It should be noted that the way we decode the symbol vector to a given iteration, is to select the symbol vector which has the lowest cost function in all the iterations up to that point in time.

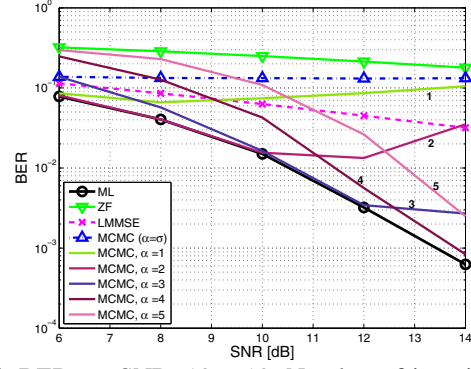


Figure 5: BER vs. SNR, 10×10 . Number of iterations, $k = 100$.

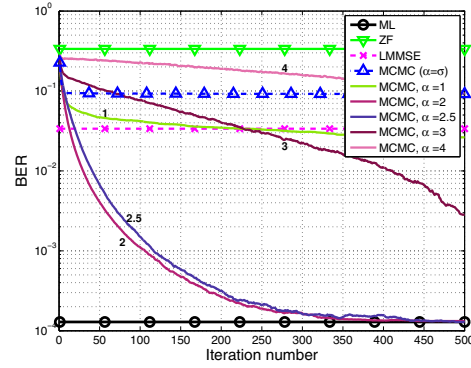


Figure 6: BER vs. iterations, 50×50 system. SNR = 12 dB.

Table I: Complexity of SD and Gibbs Sampler (GS).

| N | Method | SNR 6 dB | 10 dB | 14 dB |
|-----|--------|----------------------|----------------------|-------------------|
| 10 | GS | $9.8 \cdot 10^3$ | $10.9 \cdot 10^3$ | $16.4 \cdot 10^3$ |
| | SD | $10.0 \cdot 10^3$ | $1.7 \cdot 10^3$ | $1.5 \cdot 10^3$ |
| 50 | GS | $7.6 \cdot 10^5$ | $9.5 \cdot 10^5$ | $10.6 \cdot 10^5$ |
| | SD | $\gg 1.9 \cdot 10^9$ | $\gg 1.9 \cdot 10^9$ | $37.7 \cdot 10^5$ |

decoder has an enormous complexity under moderate SNR.¹¹ Therefore, it has not been possible to simulate the performance of this decoder within a reasonable time and we have therefore “cheated” a little by initializing the radius of the sphere to the minimum of either the norm of the transmitted symbol vector or the solution found by the heat bath. This has been done in order to evaluate the BER performance of the optimal detector. Figure 6 shows the BER curve as a function of the iteration number, while Figure 7 illustrates the BER curve vs. the SNR. From Figure 6 we see that there is a quite good correspondence between the simulated α and the theoretical value $\alpha_+ = 2.6$ obtained from Figure 2. The average complexity (MAC pr. symbol vector) of the heat bath having a BER performance comparable with the ML detector is shown in Table I. The SD has been included as a reference.¹² It is observed that the

¹¹In fact, it can be shown that, for $\text{SNR} = \mathcal{O}(\ln N)$, the lower bound on the complexity of the sphere decoder obtained in [7] is exponential.

¹²It has not been possible to simulate the SD for a 50×50 system when $\text{SNR} \leq 10\text{dB}$ and, therefore, the complexity of $\text{SNR} = 12\text{dB}$ has been used a lower bound.

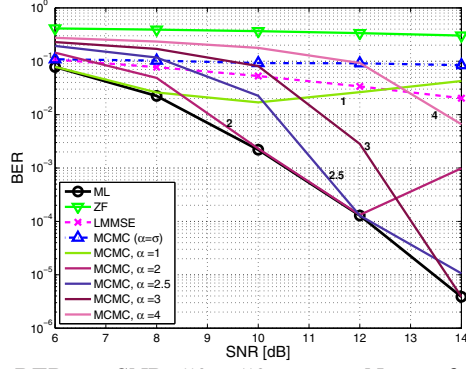


Figure 7: BER vs. SNR, 50×50 system. Num. of iter., $k = 500$.

complexity of the heat bath is not affected by the SNR as much as the SD.

VIII. CONCLUSION

In this paper we considered solving the integer least-squares problem using Monte Carlo Markov Chain heat bath. The novelty of the proposed MCMC method is that, unlike simulated annealing techniques, we have a fixed temperature parameter in all the iterations, with the property that after the Markov chain has mixed, the probability of encountering the optimal solution is only polynomial small (i.e. not exponentially small). We further compute the optimal (here largest) value of the temperature parameter that guarantees this. Simulation results indicate the sensitivity of the method to the choice of the temperature parameter and show that our computed value gives a very good approximation to its optimal value. Investigating whether the Markov chain mixes in polynomial time for this choice of temperature parameter remains open. We have identified, and briefly outlined, several lines of attack for this problem.

IX. APPENDIX

A. Proving Lemma IV.1

Lemma IV.1 (Gaussian Integral) *Let \mathbf{v} and \mathbf{x} be independent Gaussian random vectors with distribution $\mathcal{N}(\mathbf{0}, \mathbf{I}_N)$ each. Then*

$$E \left\{ e^{\eta(\|\mathbf{v} + a\mathbf{x}\|^2 - \|\mathbf{v}\|^2)} \right\} = \left(\frac{1}{1 - 2a^2\eta(1 + 2\eta)} \right)^{N/2}. \quad (23)$$

Proof: In order to determine the expected value we compute the multivariate integral

$$\begin{aligned} & E \left\{ e^{\eta(\|\mathbf{v} + a\mathbf{x}\|^2 - \|\mathbf{v}\|^2)} \right\} \\ &= \int \frac{d\mathbf{x}d\mathbf{v}}{(2\pi)^N} e^{-\frac{1}{2}[\mathbf{v}^T, \mathbf{x}^T] \begin{bmatrix} \mathbf{I}_N & -2a\eta\mathbf{I}_N \\ -2a\eta\mathbf{I}_N & (1 - 2a^2\eta)\mathbf{I}_N \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{x} \end{bmatrix}} \\ &= \frac{1}{\det^{N/2} \begin{bmatrix} 1 & -2a\eta \\ -2a\eta & 1 - 2a^2\eta \end{bmatrix}} = \left(\frac{1}{1 - 2a^2\eta(1 + 2\eta)} \right)^{N/2}. \end{aligned}$$

Thus, Lemma IV.1 has hereby been proved. \blacksquare

REFERENCES

- [1] B. Hansen, M. and Hassibi, A.G Dimakis, and W. Xu, "Near-Optimal Detection in MIMO Systems using Gibbs Sampling," in *IEEE Global Telecommunications Conference (GLOBECOM)*, 2009, pp. 1–6.
- [2] B. M. Hochwald and S. Ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. on Commun.*, vol. 51, no. 3, pp. 389–399, 2003.
- [3] B. Hassibi and H. Vikalo, "On the Sphere-Decoding Algorithm. I. Expected Complexity," *IEEE Trans. on Sig. Proc.*, vol. 53, pp. 2806–2818, Aug. 2005.
- [4] B. Hassibi and H. Vikalo, "On the Sphere-Decoding Algorithm. II. Generalizations, Second-Order Statistics, and Applications to Communications," *IEEE Trans. on Sig. Proc.*, vol. 53, pp. 2819–2834, Aug. 2005.
- [5] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Transactions on Information Theory*, vol. 48, no. 8, pp. 2201–2214, 2002.
- [6] M. O. Damen, H. E. Gamal, and G. Caire, "On Maximum-Likelihood Detection and the Search for the Closest Lattice Point," *IEEE Trans. on Info. Theory*, vol. 49, pp. 2389–2402, Oct. 2003.
- [7] J. Jaldén and B. Ottersten, "On the Complexity of Sphere Decoding in Digital Communications," *IEEE Trans. on Sig. Proc.*, vol. 53, pp. 1474–1484, Apr. 2005.
- [8] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer, 2 edition, 2004.
- [9] O. Häggström, *Finite Markov chains and algorithmic applications*, Cambridge University Press, 2002.
- [10] H. Zhu, B. Farhang-Boroujeny, and R.R. Chen, "On performance of sphere decoding and Markov chain Monte Carlo detection methods," *IEEE Sig. Proc. Letters*, vol. 12, pp. 669–672, 2005.
- [11] B. Farhang-Boroujeny, H. Zhu, and Z. Shi, "Markov chain Monte Carlo algorithms for CDMA and MIMO communication systems," *IEEE Trans. on Sig. Proc.*, vol. 54, no. 5, pp. 1896–1909, 2006.
- [12] X. Wang and V. H. Poor, *Wireless Communications Systems: Advanced Techniques for Signal Reception*, Prentice Hall, 2003.
- [13] R. Chen, JS Liu, and X. Wang, "Convergence analyses and comparisons of Markov chain Monte Carlo algorithms in digital communications," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 255–270, 2002.
- [14] D.J.C. MacKay, *Information Theory, Inference and Learning Algorithms*, Cambridge University Press, 2003.
- [15] S. A. Laraway and B. Farhang-Boroujeny, "Implementation of a Markov Chain Monte Carlo Based Multiuser/MIMO Detector," *IEEE Trans. on Circuits and Sys. - I: Regular Papers*, vol. 56, pp. 246–255, 2009.